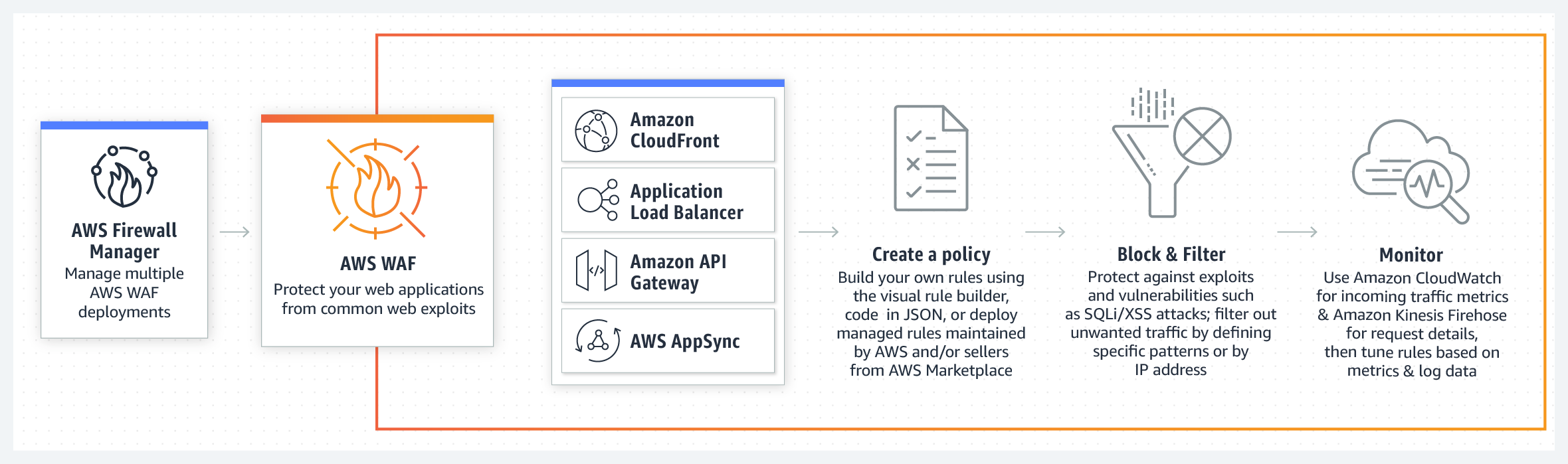
**AWS Web Application Firewall Control Web Traffic using Web Application Firewall**



**INTRODUCTION**

**WAF (Web Application Firewall) :**

* AWS WAF is a web application firewall that helps protect your web applications or APIs against common web exploits and bots that may affect availability, compromise security, or consume excessive resources
* AWS WAF gives you control over how traffic reach your application by enabling you to create security rules thats blocks common attack pattern like **SQL injection** and **cross-site scripting.**
* It only allow s the request to reach the seerver based on the rules or pattern you define.
* User creates thier own own rules and specifiy the conditions that AWS WAF searches for incomming web request s.
* The cost of WAF is only for what you use.
* **The pricing is based on how many rules you deploy and how many web requests your application receives.**

For example you can deplo AWS WAF on CloudFront with an Application  
Load Balancer in front of your web servers or servers running on EC2.

**Feature of WAF**

* Web traffic filtering using custom rules.
* You can create your own rules, depending on your requirements , whether to block or allow the incomming and outgoing request. You can also customize the string that appears in your web request.
* Blocking malicious request.
* You can also configure rules in AWS WAF to identify and block web requests threads like SQL injection & cross-site scripting .
* Tune your rules & monitor traffic.
* AWS WAF also allow us to review aur rules and customize them to prevent new attacks from reaching the server.

**Activity Guide Description :**

**Application Load Balancer (ALB)**

* Load Balancer a service that allow you to destribute the incomming application or network traffic across multiple targets, such as Amzon EC2 instances,

containers, and IP addresses in multiple Availablity Zones (AZ).

* ALB is used to route the HTTP & HTTPS traffic accross the targets based on the rules attached with the target group.
* Rules determine what action is taken wehn a rules matchs a client request
* The target group is used to route requests accross registered targets as part of an action rules. Target group consist a protocal & target port we can also configure health check to monitor the status of the target group. A single ALB can route traffic to multiple target group.
* Targets consist of EC2 instances that are registerd with the ALB as part of a target group.

**Web Servers**

* Two Web Server are launched in the Subnet to handle the web requset.
* The request to a web server is shared using the ALB
* Web Server attached to the ALB Target Group.
* Server are installed with HTTPD or both servers & have the test pages RESPONSE COMING FROM IT Team (**Instance A**) & RESPONSE COMING FROM SERVER 2 (**Instance B**) respectively.
* They are attached to a security group via port 80 that allows the web traffic comming from ALB.

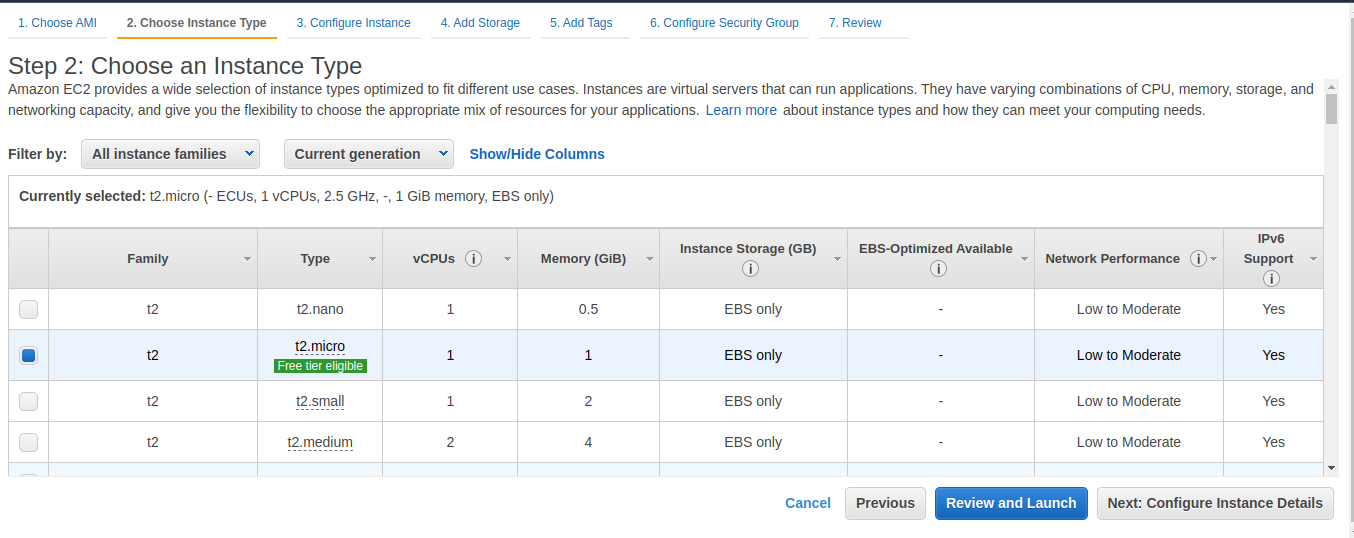
**LAUNCH A WEB SERVER (INSTANCE A) IN ONE OF THE AVAILABLITY ZONES**

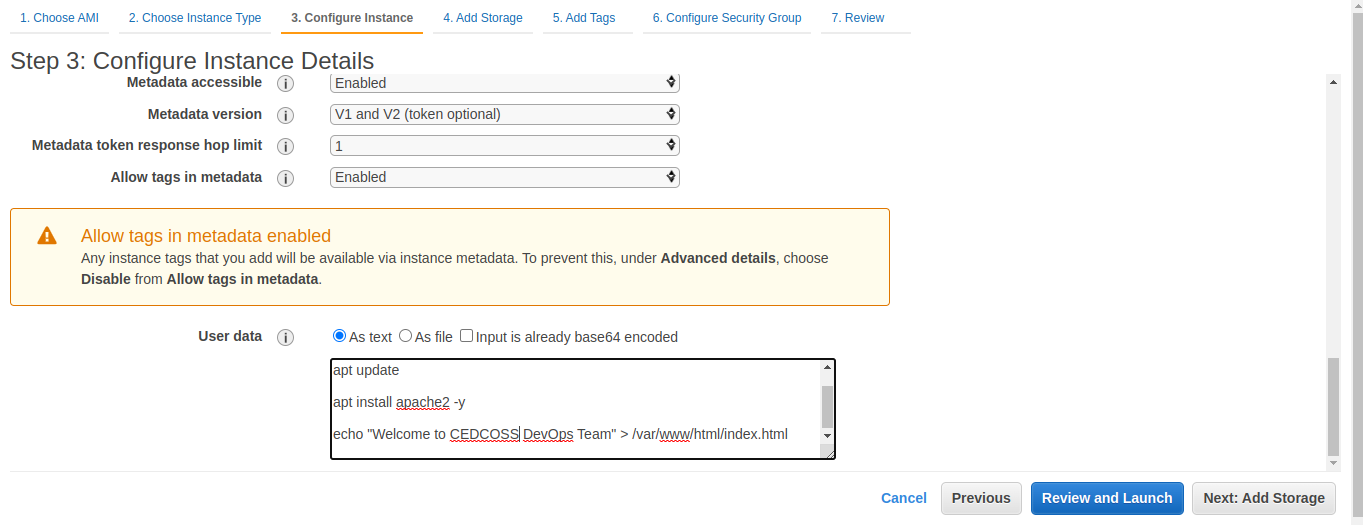
In this section you will launch two Ubuntu EC2 Instances with an Apache web server & basic application installed on initialization. You will also demonstrate a simple example of bootstraping using AWS EC@ metadata services.

**Note** Make sure you perform in one region. We are using **US east (N. Virginia) us- east-1**

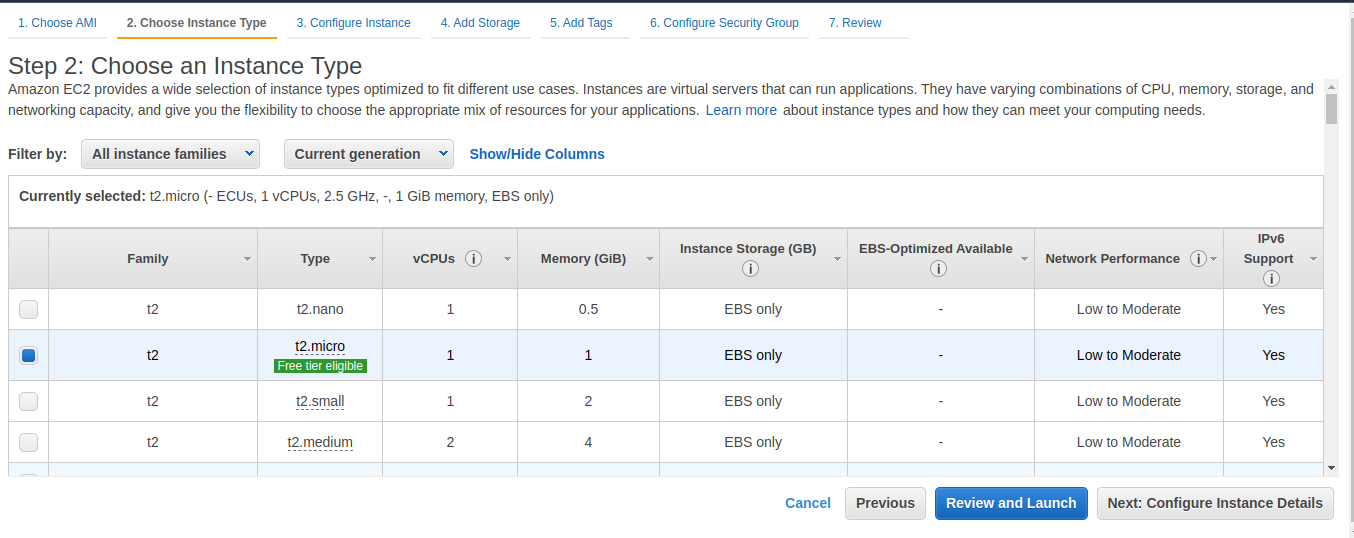
**Control Web Traffic Using : Web Application Firewall**

Go to AWS console ----> EC2 ------> Create two EC2 instances ---> “WebServer1”

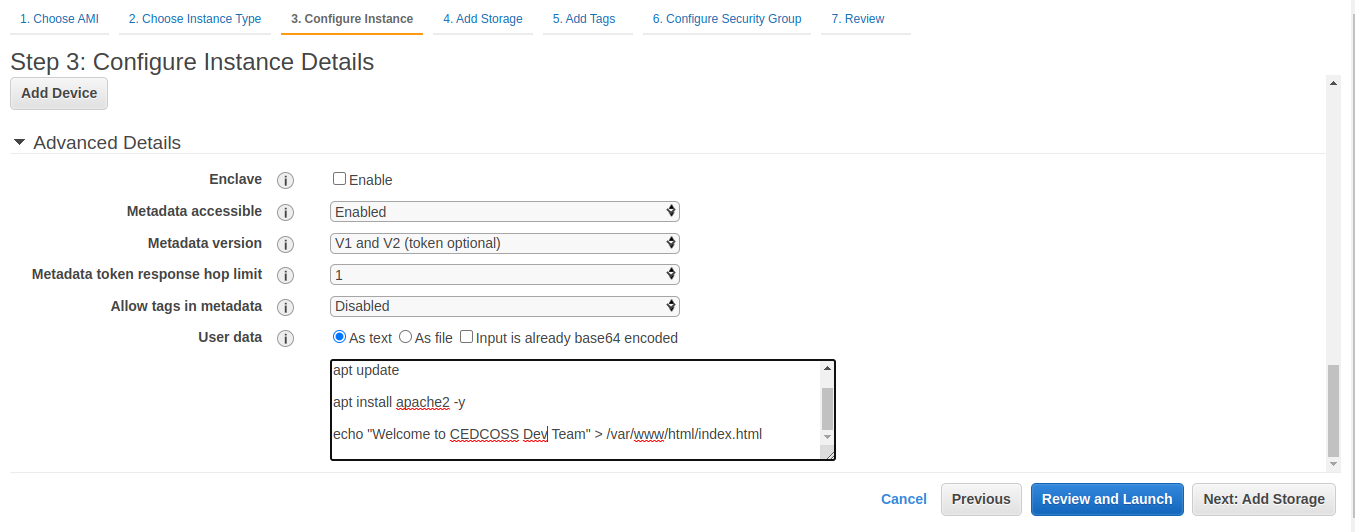
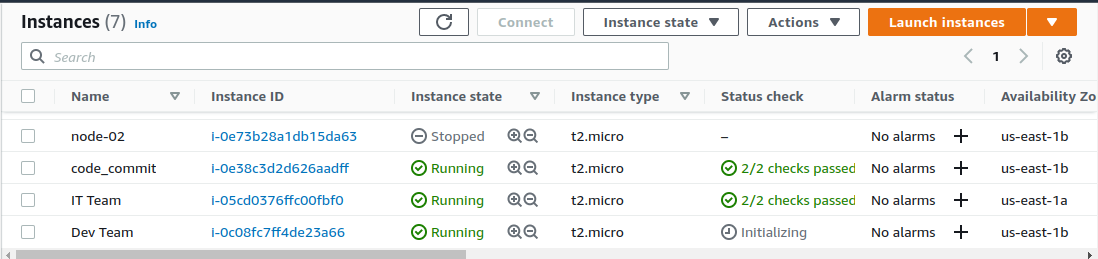
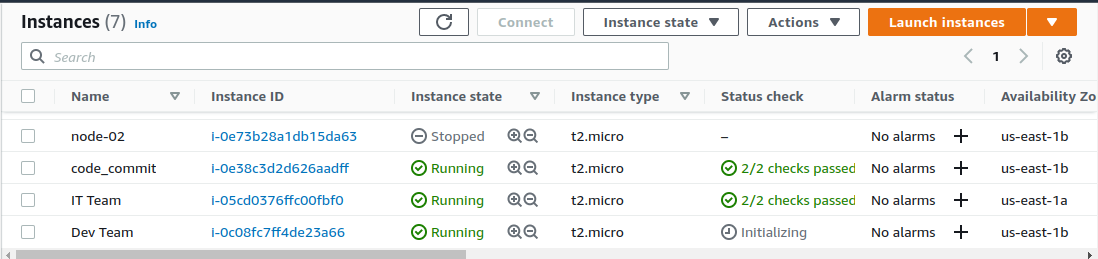
 Install Apche web server. **“Make sure choose same Security Group for both instances”**



Go to AWS console ----> EC2 ------> Create two EC2 instances ---> “WebServer2”



Install Apche web server. **“Make sure choose same Security Group for both instances”**

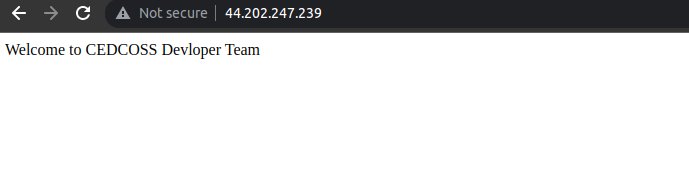
**** 

“One both instances are running take public IP & paste on the browser & hit enter”

**IT Team :**

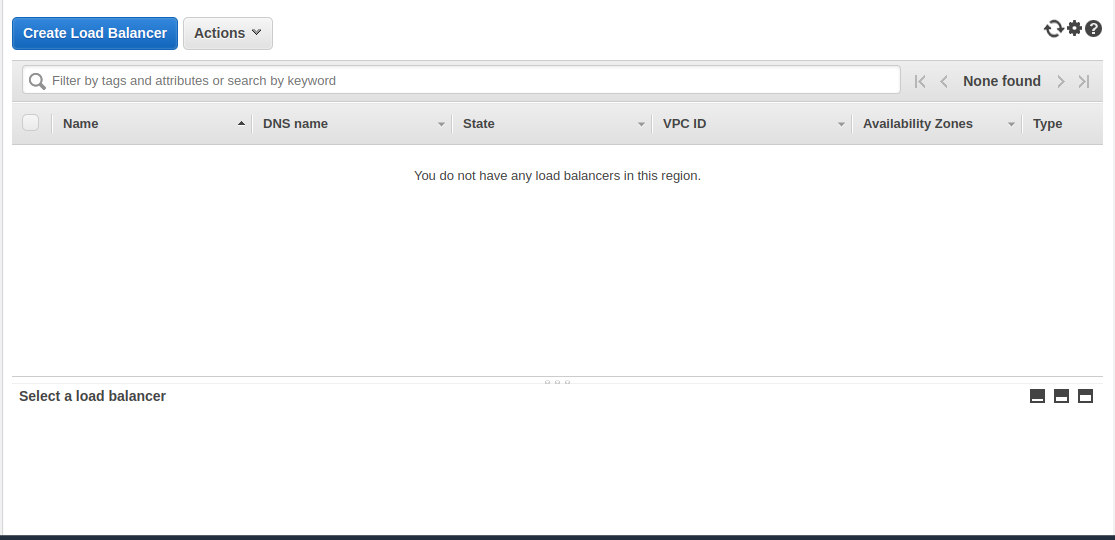


**Dev Team :**

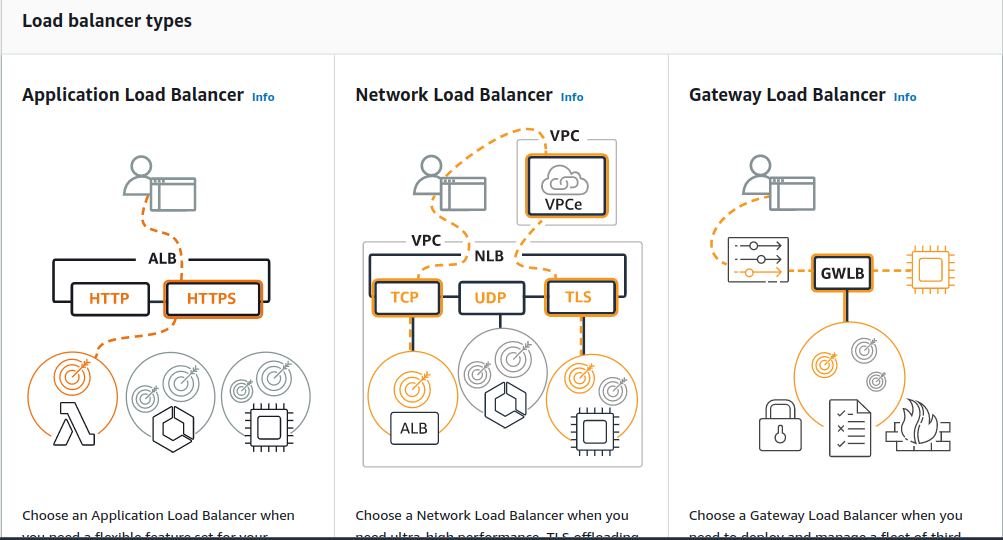


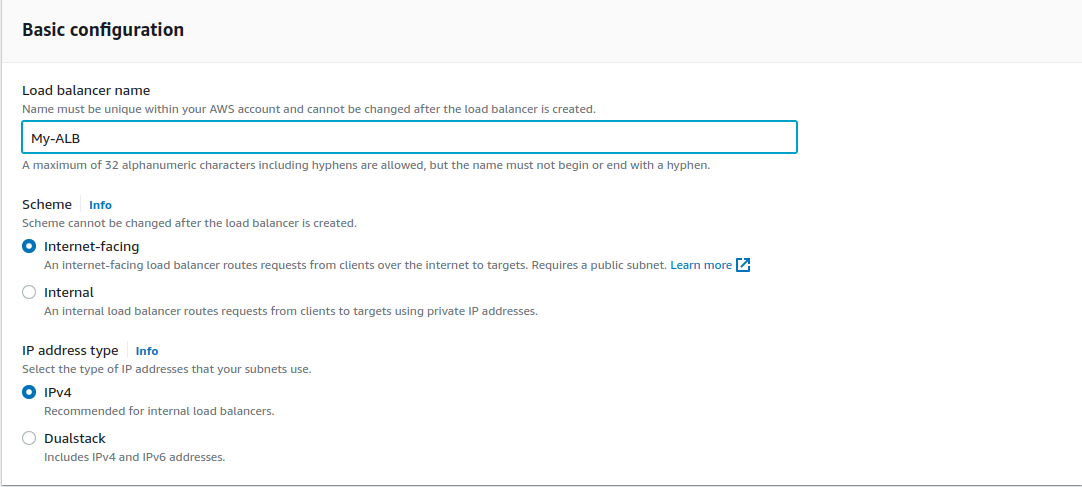
**Load Balancer :**

Open AWS console ---> EC2 ---> Load Balancer ---> Click on Load Balancer

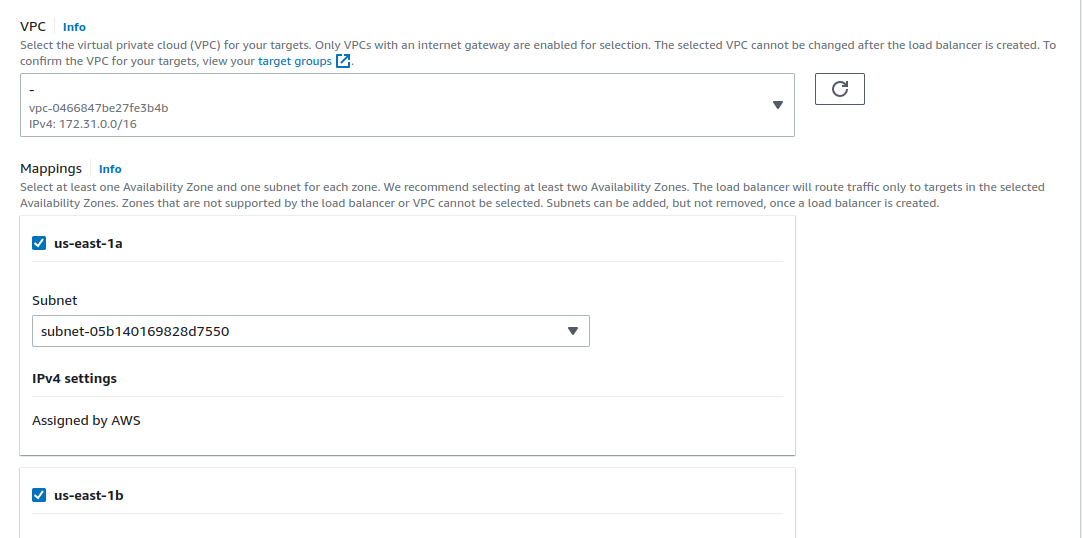


Under Load Balancer, Click on create. **ALB**

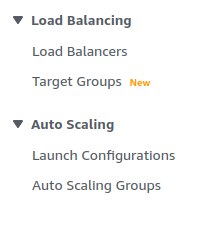




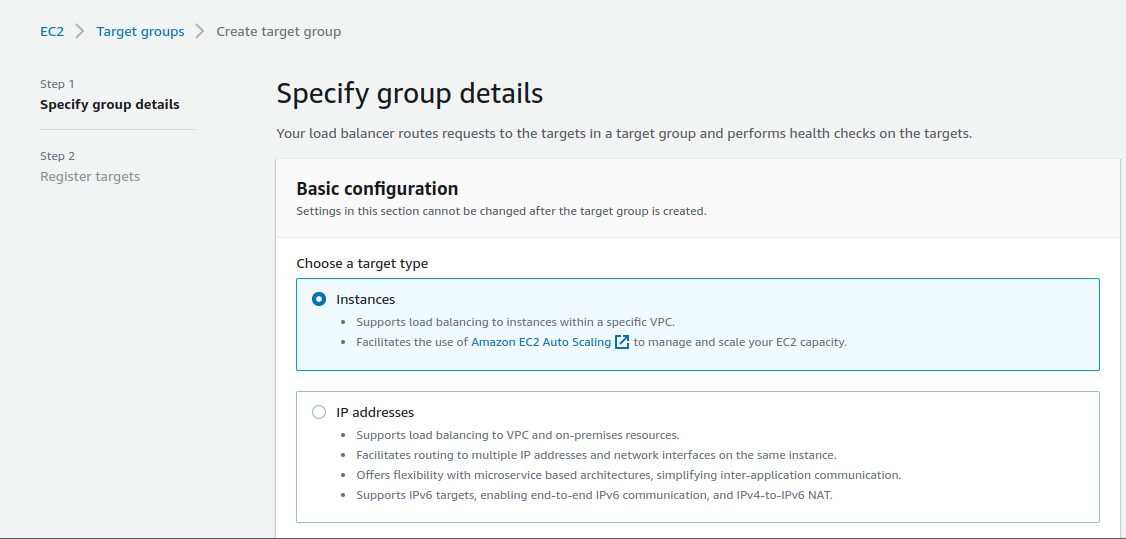
Under Mapping select your Availablity Zone Subnet whcih you create in your

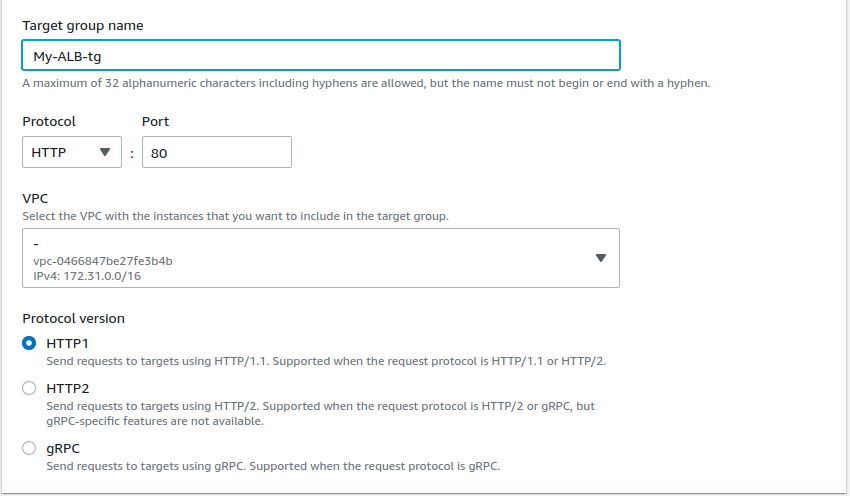
Instances.

Stay this page open next tab ---> open same console --> EC2 ----> Target Group.

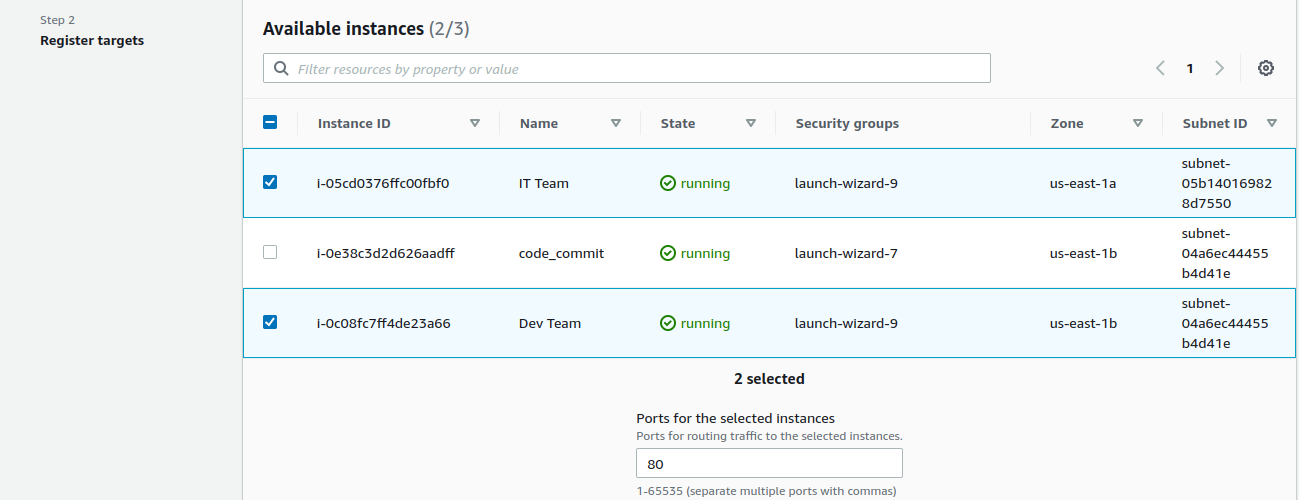


Create Target Group ----> Select instance level -----> Enter the name of group

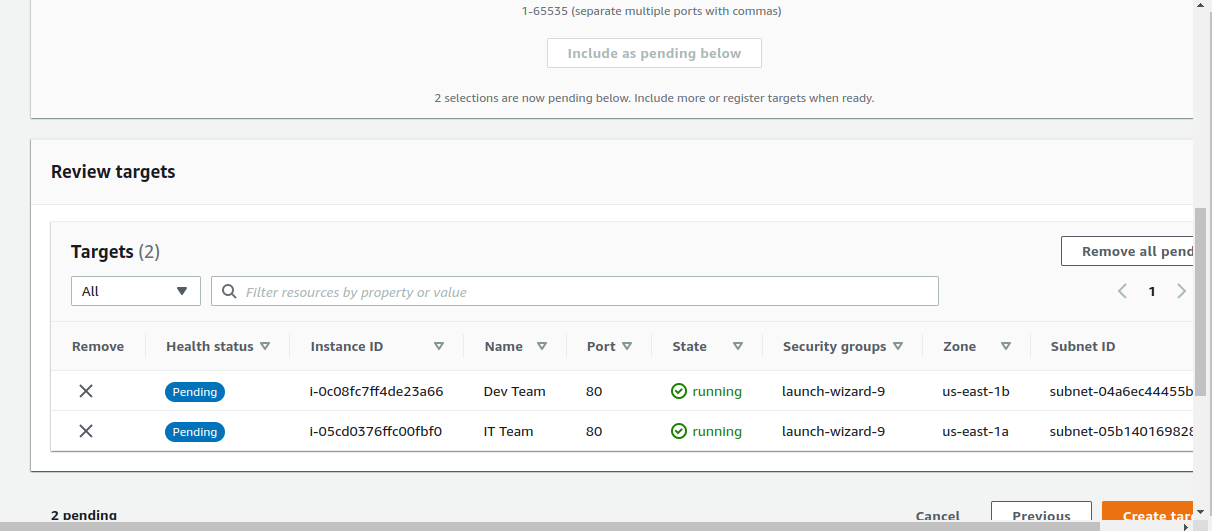




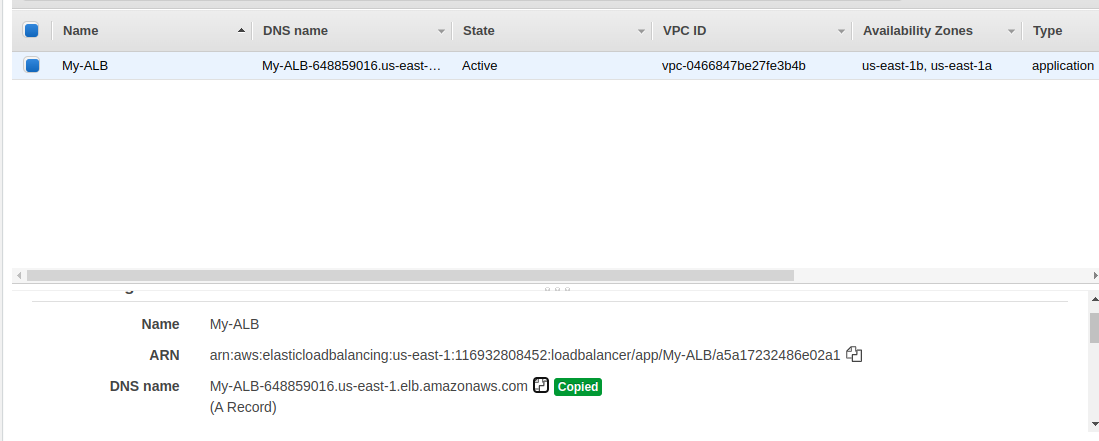
Click next ---> Register target instances select them.



Click on **Include as pending below** ---> Click **Create target group**



Back to Load Balancer Done the remain things -----> click on create Load Balancer.



Copy DNS Name & hit on browser.

You shall see one of the web server either **CEDCOSS DevOps Team** OR

**CEDCOSS Devloper Team** we are getting as responseas the Load Balancer is routing the traffic (change at every click).

**Create a AWS WAF**

WAF will have IP set (Collection of IP Addresses) and Web ACL that defines rules to grant/deny access and use IP Set.

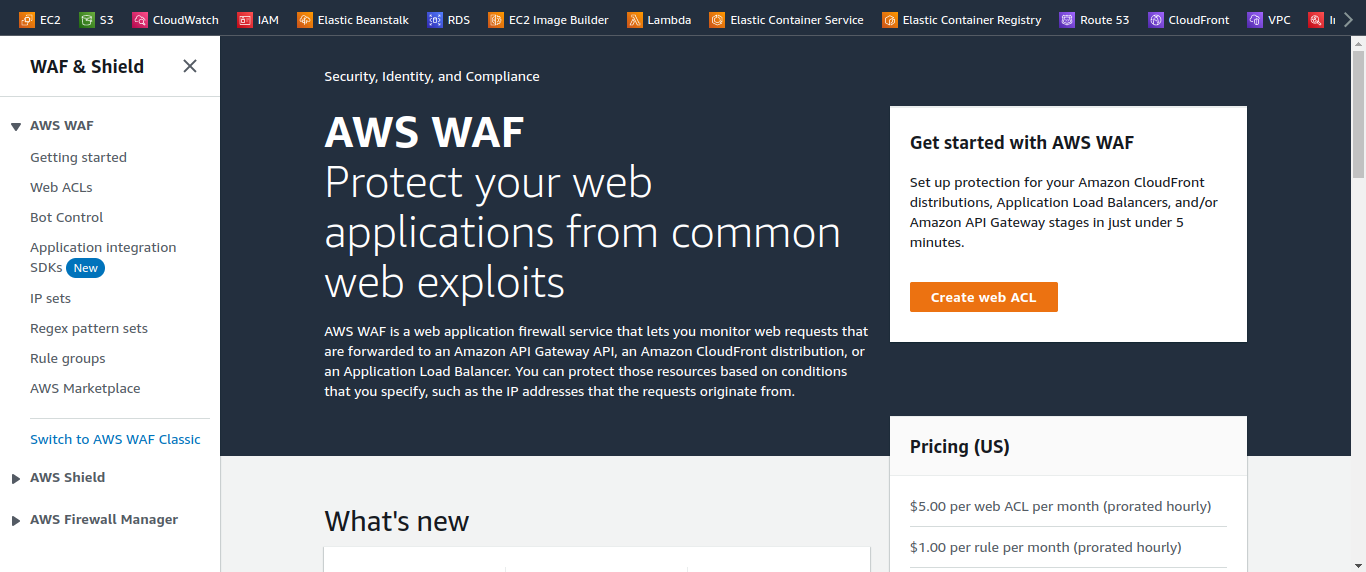
**Create an IP set**

An IP sets provides a collection of IP addresses & IP address ranges that you want to use togather in a rule statemet. IP setes are AWS resources.

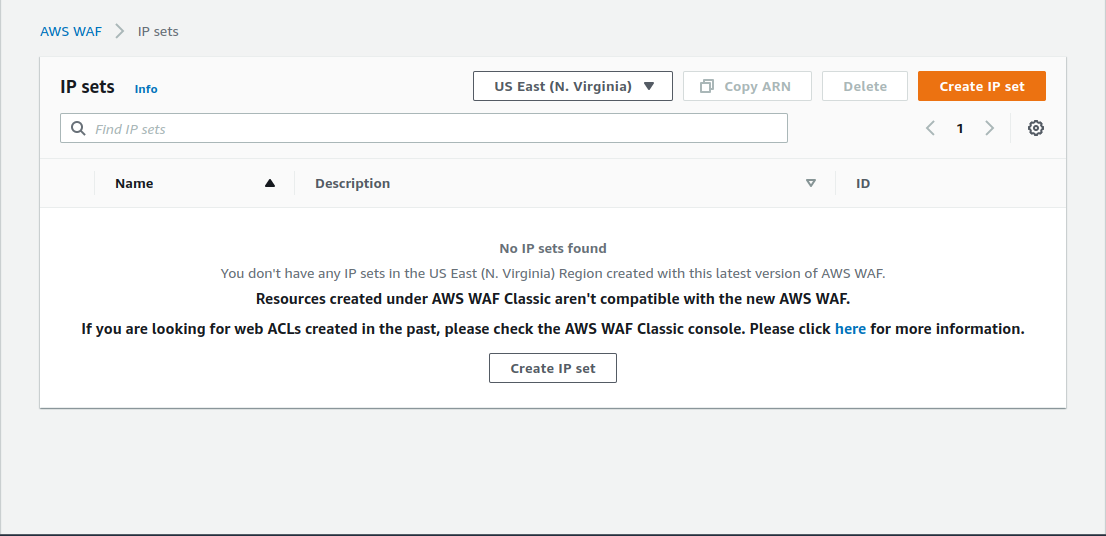
To use an IP set in a web ACL or rule of group, you first create an AWS resouce, IPSet with your address specification. Then you refrence the set when you add an IP rule statement to a

web ACL or rule group .

Click on search bar to top & select WAF ----> Open it new tab



Select IP Set ---> Click on create IP Set.



**IP Set Details :**

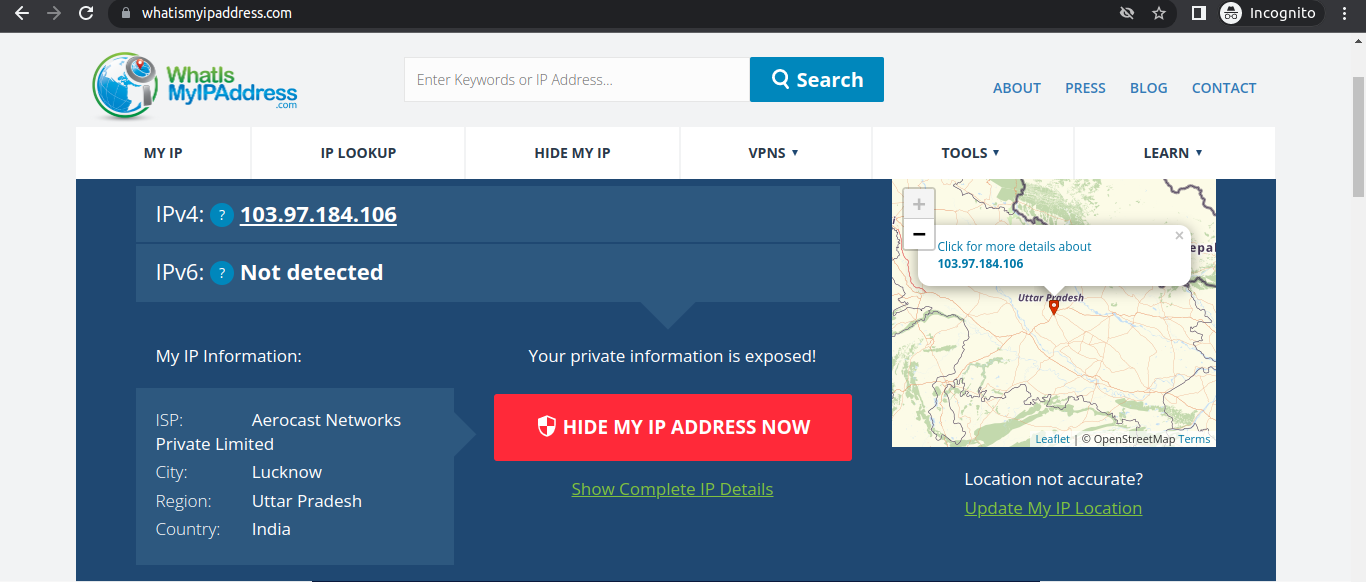
**- IP Set Name :** MyIPSet

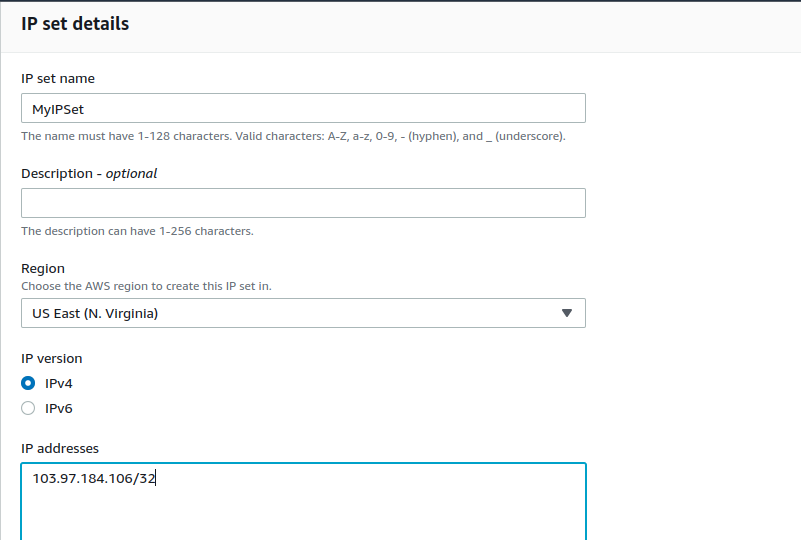
**- Description :** IP set to block my public IP

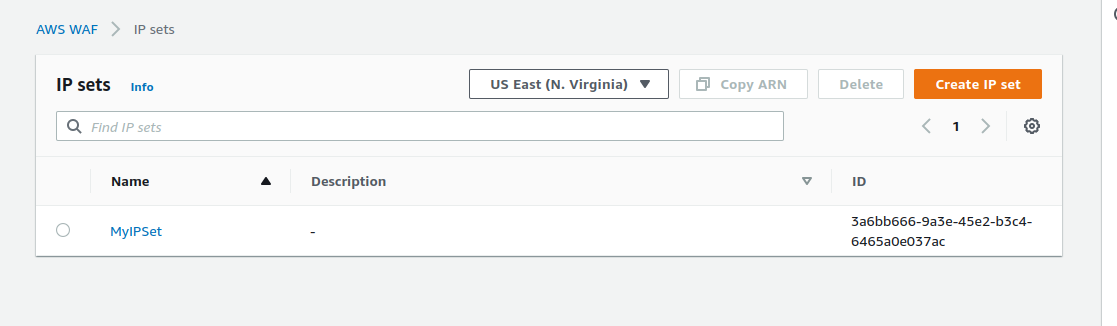
**- Region :** US EAST (N. VERGINIA)

**- IP Version :** IPv4

**- IP Address :** Ip Address of you local network/32 (search on google ‘whatismyip’)



**Note :** You have to give /32 after the IP is pasted or else you won’t be able to create an IP set. Here /32 is CIDR range considering of single IP i.g. Public IP assigned to your hsot



**Create a Web Access Control List (Web ACL):**

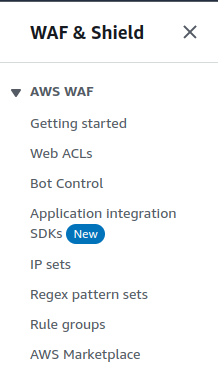
A **Web Access Control List** (Web ACL) gives you fined-grained control over the web requests that your protected resource responds to, you can protect Amazon CloudFront, Amazon API Gateway, Application Load Balancer, & AWS AppSync resources.

You can use criteria like the following to allow or block requests :

* IP address origin of the request
* Country of the origin of the request
* String match or regular expression (regex) match in a part of the request
* Size of a prticular part of the request
* Detection of malicious SQL code or scripting

**Web ACL Details**

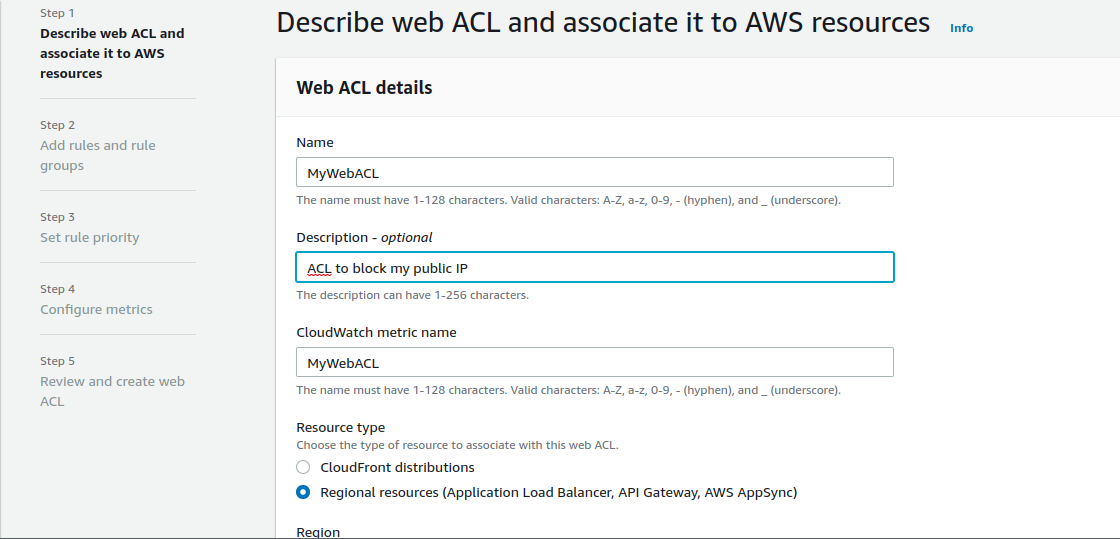
* Navigate to the AWS WAF dashboard and select Web ACL ----> Click on create Web ACL to create a new web ACL



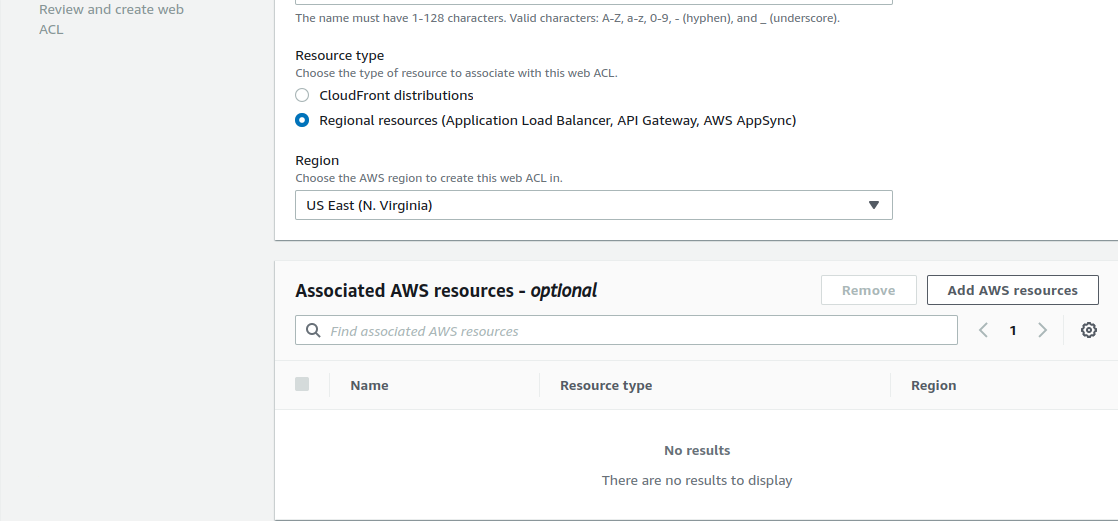
* Configure ACL as below :

**Web ACL Details**

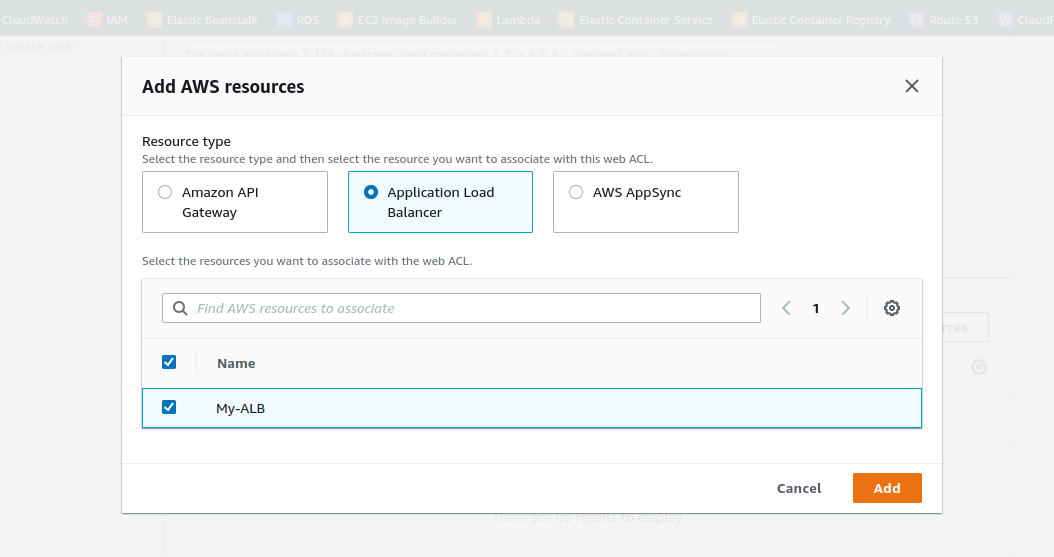
* Name : MyWebACL
* Description : ACL to block my public IP
* CloudWatch matric name : MyWebACL
* Resource type : Regional resouces (Appllication Load Balancer, API Gateway, AWS AppSync).
* Region : US East (N.Virginia)



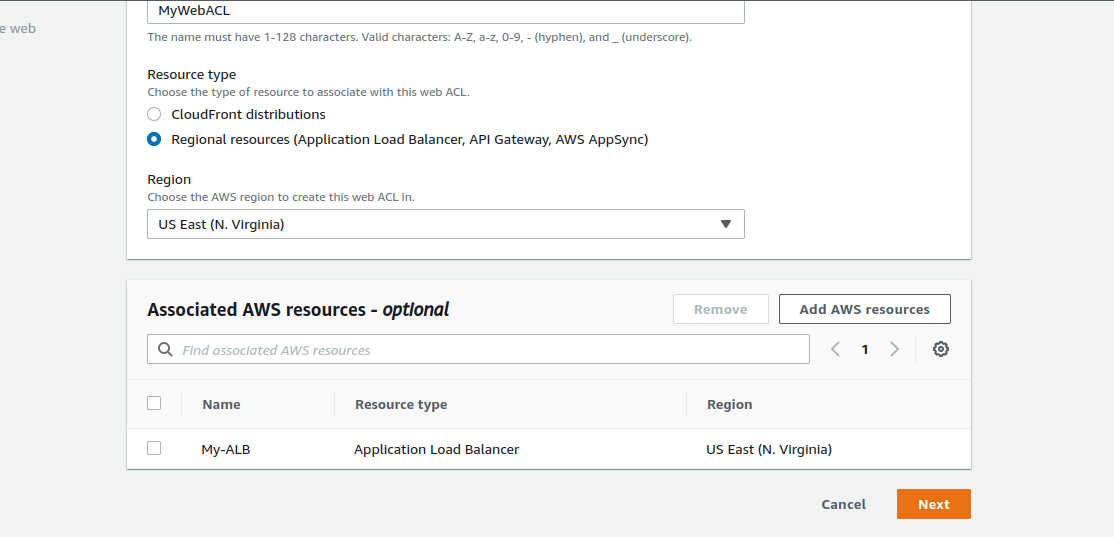
To associate an AWS resources click on **Add AWS Resources**



* In Add AWS Resource select **Applicatopn Load Balanacer** and select Name of **ALB**. Click on **Add**.

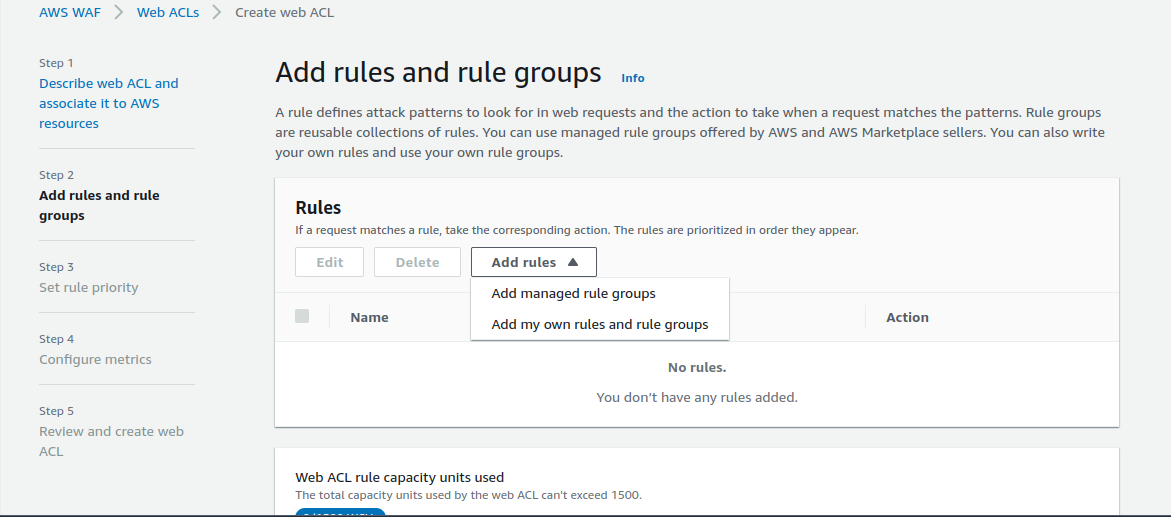


Lastly click on the next.



**Add rules & rule groups :**

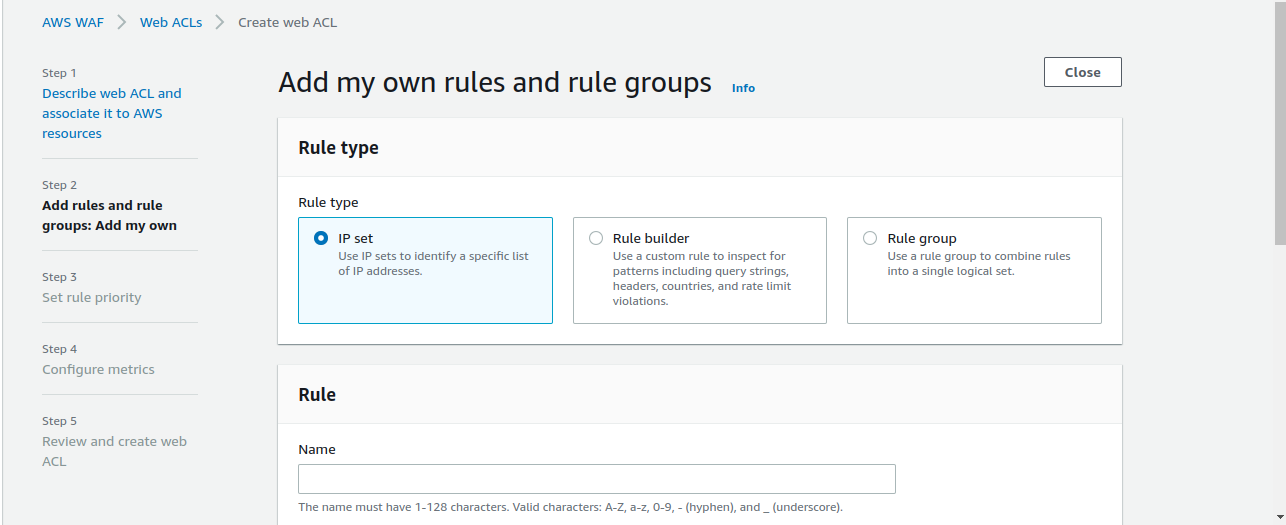
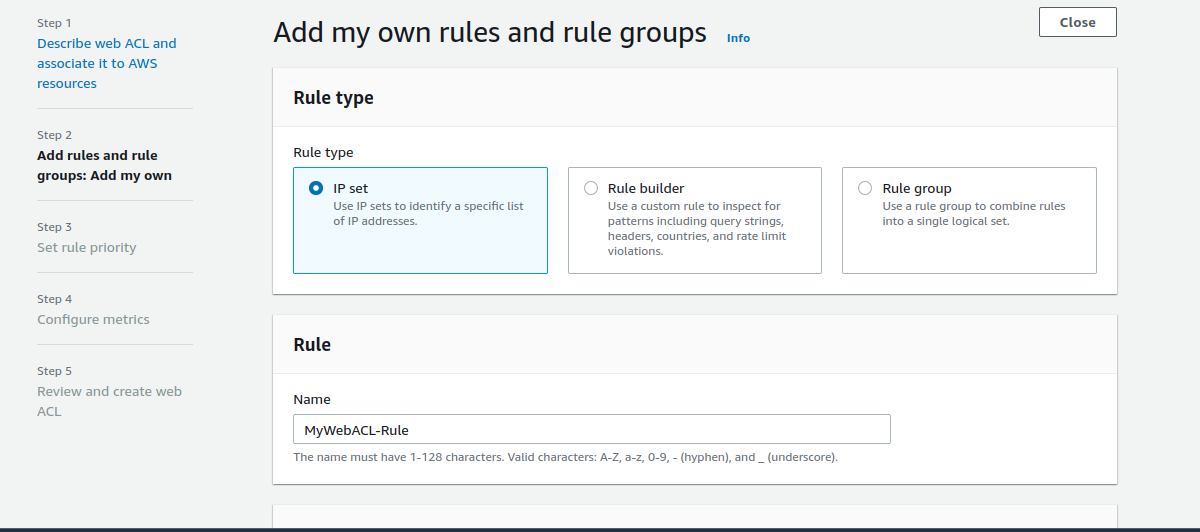
* Under **Rules** click on **Add rule**  & select **Add my own rule & rule group** in the drop down menu.

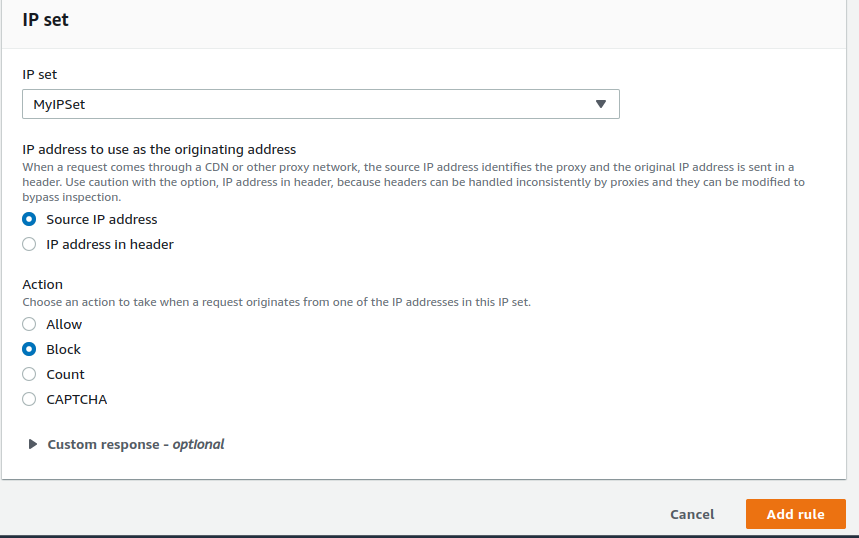


* In **Rule type IP set as** shown below & fill the details as given below :
* Rule type : IP sets (**Use IP sets to identify a specific list of IP addresses.**)
* Name : **MyWeb ACL-Rule**
* IP Sets : select the IP set created Above (**MyIPSet**)
* IP Addresss to use as the originating address : **Source IP address**

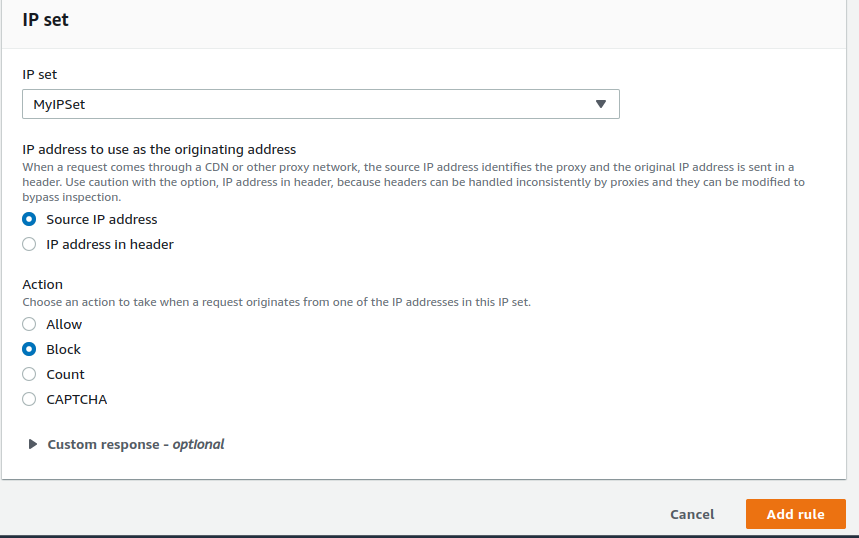
**Note :** When a request comes throgh a CDN or other proxy network, the source IP address identifies the proxy and the original IP address is sent in a header. Use caution with the option, IP address in header, because headers can be handled inconsistently by proxies & they can be modified to bypass inspection.

* Action **: Block**

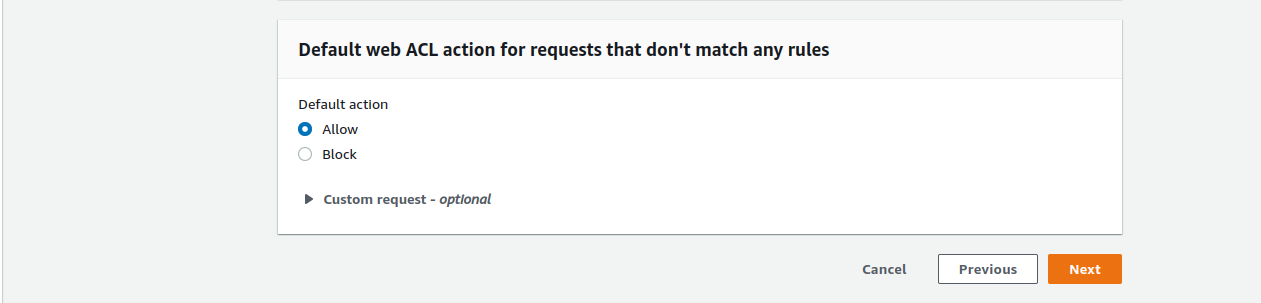
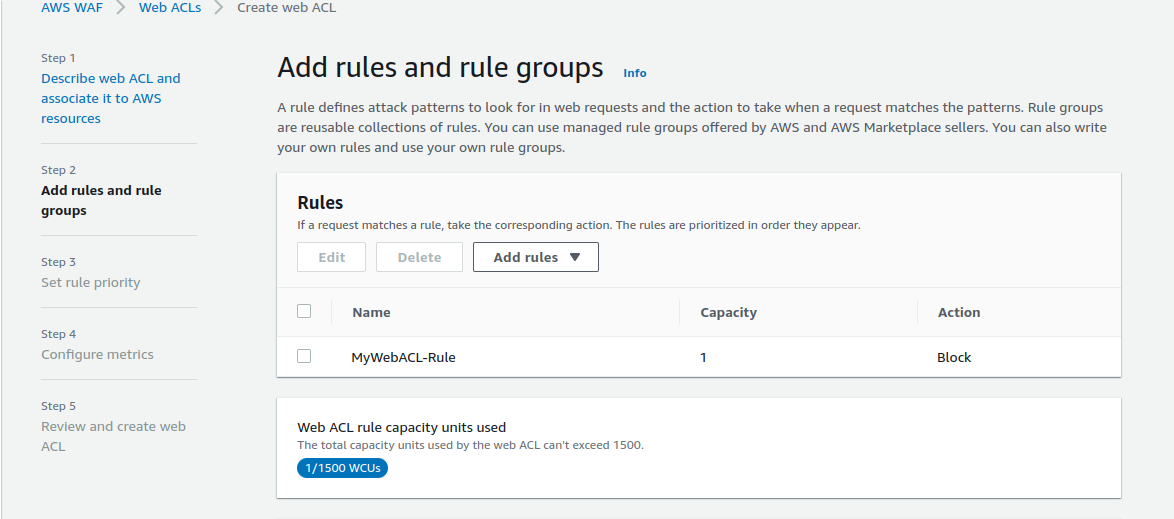
**** 



* Onece you provide above details, click on **Add rule.**



**Lastly click on Next.**



**Set rule priority**

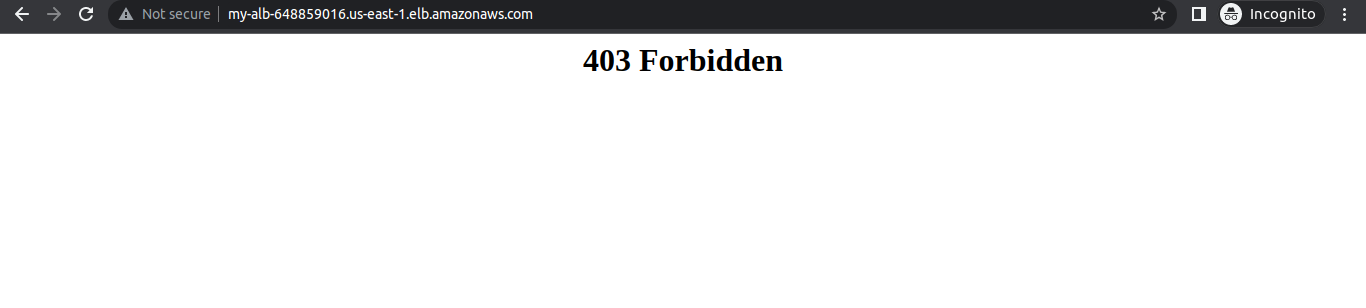
AWS WAF evaluate the rules & rule groups in the order shown, starting from the top.

Move rules up or down to change the evaluation order but we have only one rule, so we don’t need to change

* Leave as default & click on **Next.**

**TESTING THE WORKING OF THE WAF :**

* To test the WAF, navigate to **Load Balancer** from EC2 left menu under the sub – heading **Load Balancing**
* Under the Load Balancer section select the Application Load Balancer My-ALB
* Copy the DNS name under Description & paste it in your desired browser
* **Example :** http://my-alb-648859016.us-east-1.elb.amazonaws.com/

Done!